Closed Form Transient Solution of Continuous Time Markov Chains Through Uniformization

Llorenç Cerdà-Alabern
Univ. Politècnica de Catalunya, Computer Architecture Dep.
Barcelona, Spain
Email: llorenc@ac.upc.edu

***Abstract*—This paper derives a mapping between a close form transient solution of a continuous time Markov chain (CTMC), and a close form transient solution of one of its uniformized discrete time Markov chains. This result is used to propose a new method to compute the transient solution of CTMCs. The method is simple to implement and has important benefits: (i) It yields a closed-form solution. (ii) It can be used to compute the transient solution for general CTMCs (even with absorbing states, or defective matrices). (iii) It can be used even for chains having a large number of states.**

***Index Terms*—Markov chains, transient solution, Eigenvectors method, Vandermonde system, Uniformization.**

I. Introduction

Probabilistic modeling using Markov chains has been successfully exploited in almost all fields of modern applied mathematics, and particularly in computer communications. The *stationary* solution of a Markov chain is easier to compute than the *transient* solution, and it is enough in many cases. However, some applications, as reliability modeling [1], multiprocessor load balancing [2], network survivability [3] and others are primarily interested in the transient solution. Many methods have been proposed to compute the transient solution of Markov chains. Some examples are the approaches based on Laplace transform techniques [1], [4], the exponential matrix [5], finite-differencing [6], differential equation solvers [7], Markov fluid models [8], etc. The chapter 8 of the classic book of Stewart [9] is dedicated to this topic.

This paper focuses on the class of methods based on the *undetermined coefficients* approach. This approach consists on making an intelligent guess of a closed-form expression with constant coefficients for the solution of an equation, and solve for these coefficients using boundary conditions. The undetermined coefficients approach has been successfully used to find the solution of many difference and differential equations (see e.g. [10]). The transient solution of a Discrete or Continuous Time Markov Chain (DTMC and CTMC), is the solution of a difference or differential equation respectively. Therefore, they are suited to be computed using algorithms based on the undetermined coefficients approach.

The well known Eigenvectors method [5], explained later in this paper, belongs to the class of undetermined coefficients approaches. Another possibility is applying the undetermined coefficients approach directly to the solution of the Markov chain, expressed in terms of its eigenvalues, as done e.g. in [11]. By doing so, the undetermined coefficients are obtained by solving a Vandermonde system of equations. For this reason thorough the paper this method will be referred to as the *Vandermonde method*. In this paper the pros and cons of these techniques are analyzed. Furthermore, we derive the mapping between the undetermined coefficients of a CTMC and those of one of its uniformized DTMCs. This result is of theoretical interest, and to the best of our knowledge, this is the first time that such mapping is established. We use this result to propose a novel approach to compute a close form transient solution of a CTMC. Thorough the paper, this novel proposal will be referred to as the *Uniformized Vandermonde* (UV) method. Numerical results show that it yields a simple yet powerful technique to compute a close form transient solution of a CTMC.

Before introducing the Uniformized Vandermonde method the paper dedicates sections II to V to review the solution of DTMC and CTMC using the undetermined coefficients approach. Most of the contents of these sections are not a contribution of this paper. For instance, the Eigenvectors method applied to Markov chains can be found in classic books as [12, sec. 4.8]. The expression for the exponential matrix having confluent eigenvalues (equation (14) in section IV) can be found in previous works [13], [14], [15], [7]. Nevertheless, to our best knowledge, the equivalent equation for a DTMC (equation (2), section II), has been not previously reported in the literature. The CTMC counterparts have been included for the sake of completeness, because they are the basis of Uniformized Vandermonde method presented in section VI, and because they are used in the numerical results presented in later sections. An additional reason is that, even if the Vandermonde method is one of the simplest ways to compute the transient solution of a Markov chain, it is not covered, or it is done superficially, in the books found in the literature. This paper tries to fill this gap, highlighting the striking parallelism that exists when this method is used to compute the transient solution of DTMCs and CTMCs.

The rest of the paper is organized as follows. First, the function with constant coefficients for the transient solution of a DTMC is presented in section II, and in section III the coefficients are computed using the Eigenvectors and Vandermonde methods. In sections IV and V, the same it is done for a CTMC. The Uniformized Vandermonde method is presented in section VI, in section VII numerical experiments

are carried out analyzing the different methods described in this paper. Section VIII discuss the stability of the Uniformized Vandermonde method. Finally, concluding remarks are given in Section IX.

II. Discrete Time Markov Chains

Let $X(n)$ be a homogeneous finite-state discrete-time Markov chain with k states, and one step transition probability matrix of size k, $\mathbf{P}^{k\times k}$. The transient solution is given by the powers of $\mathbf{P}$ [12]:

$$\boldsymbol{\pi}(n) = \boldsymbol{\pi}(0)\,\mathbf{P}^n,\ n \geq 0 \tag{1}$$

where $\boldsymbol{\pi}(0)$ is the initial distribution, and the components of the row vector $\boldsymbol{\pi}(n)$ are the probabilities

$$\pi_j(n) = Prob\left\{X(n) = j \ \middle|\ \boldsymbol{\pi}(0)\right\},\ 1 \leq j \leq k,\ n \geq 0$$

If we are interested in computing $\pi_j(n)$ for only a reduced number of states, the following expression may be more efficient that computing the powers of the matrix $\mathbf{P}$.

Theorem 1 *Let λ_l, $l = 1, \cdots L$ be the eigenvalues of $\mathbf{P}$, each with multiplicity k_l ($k_l \geq 1$, $\sum_l k_l = k$). Without loss of generality, assume a possible eigenvalue $\lambda_1 = 0$ with multiplicity k_1. Then:*

$$\pi_j(n) = \sum_{m=0}^{k_1-1} a_j^{(1,m)}\delta_{n-m} + \sum_{l=2}^{L}\lambda_l^n \sum_{m=0}^{k_l-1} a_j^{(l,m)}\, n^m,$$
$$1 \leq j \leq k,\ n \geq 0 \tag{2}$$

where $a_j^{(l,m)}$ are constants that thorough the paper will be referred to as undetermined coefficients, and δ_k is the Kronecker's delta ($\delta_k = 1$ for $k = 0$ and 0 otherwise).

The proof of equation (2) is skipped for the sake of space.

Remark 1 *Assume that the Markov chain has the eigenvalue $\lambda_2 = 1$ with multiplicity $k_2 > 1$. Clearly, the polynomial in equation* (2) *corresponding to this eigenvalue:*

$$p_j^{(2)}(n) = \sum_{m=0}^{k_2-1} a_j^{(2,m)}\, n^m \tag{3}$$

must have the coefficients $a_j^{(2,m)} = 0$, for $m = 1, \cdots, k_2 - 1$. Otherwise, the equation (2) *will be unbounded (the polynomial* (3) *would diverge when $n \to \infty$). Therefore, the eigenvalue $\lambda_2 = 1$ has only 1 undetermined coefficient (i.e. the polynomial* (3) *corresponding to $\lambda_2 = 1$ is $p_j^{(2)}(n) = a_j^{(2,0)}$). A consequence of this fact is that the geometric multiplicity of the eigenvalue $\lambda_2 = 1$ must be equal to its multiplicity*[1]*, k_2. There may be other eigenvalues with geometric multiplicity larger than 1. However, it is not necessary to compute the geometric multiplicity of the eigenvalues: if the geometric multiplicity of an eigenvalue λ_l is g_l, when solving for $a_j^{(l,m)}$ it will be obtained that $a_j^{(l,m)} = 0$ for $m > k_l - g_l$.*

[1]The geometric multiplicity is defined as the number of linearly independent eigenvectors associated with the eigenvalue. Thorough the paper we shall use the term *confluent* to refer to eigenvalues having a multiplicity larger than their geometric multiplicity.

III. Undetermined Coefficients of a DTMC

In equation (2) there are up to k undetermined coefficients $a_j^{(l,m)}$ of $\pi_j(n)$ to be determined. This paper focuses on the computation of these coefficients. Thorough the paper the sentence *the UC of $\pi_j(n)$*, or simply *the UC*, will be used to refer to the undetermined coefficients $a_j^{(l,m)}$ of $\pi_j(n)$. Additionally, to denote the UC of $\pi_j(n)$ it will be used the notation:

$$\boldsymbol{u}_j = \begin{bmatrix}\boldsymbol{u}_j^{(1)} & \cdots & \boldsymbol{u}_j^{(L)}\end{bmatrix}^{\mathrm{T}}, \tag{4}$$

where $\boldsymbol{u}_j$ is a column vector with $\boldsymbol{u}_j^{(l)} = \begin{bmatrix}a_j^{(l,0)} & \cdots & a_j^{(l,k_l-1)}\end{bmatrix}$, $l = 1, \cdots L$ (thorough the paper we shall use $()^{\mathrm{T}}$ as the transpose operator). In the following two methods to compute the UC of $\pi_j(n)$ are described.

A. Eigenvectors Method

If the matrix $\mathbf{P}$ is diagonalizable, the Jordan blocs are reduced to scalars: $\mathbf{J}_l = \lambda_l$, and the matrix $\mathbf{P}$ admits the spectral decomposition: $\mathbf{P} = \mathbf{L}^{-1}\,\boldsymbol{\Lambda}\,\mathbf{L}$, where $\boldsymbol{\Lambda}$ is the diagonal matrix $\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1, \cdots \lambda_k)$, and $\mathbf{L}$ is a matrix whose rows, $\mathbf{l}_1, \cdots \mathbf{l}_k$, are the left-hand eigenvectors of $\mathbf{P}$. Since $\boldsymbol{\Lambda}^n = \mathrm{diag}(\lambda_1^n, \cdots \lambda_k^n)$, substituting into equation (1) we have

$$\boldsymbol{\pi}(n) = \boldsymbol{\pi}(0)\,\mathbf{P}^n = \boldsymbol{\pi}(0)\,\mathbf{L}^{-1}\,\boldsymbol{\Lambda}^n\,\mathbf{L} = \sum_{i=1}^{k} a_i\,\lambda_i^n\,\mathbf{l}_i \tag{5}$$

where we assume $\lambda_i^n = \delta_n$ for any $\lambda_i = 0$. Defining the row vector $\mathbf{a} = \begin{bmatrix}a_1 & \cdots & a_k\end{bmatrix}$, the constants a_i can be obtained solving the system of equations:

$$\mathbf{a}\,\mathbf{L} = \boldsymbol{\pi}(0) \tag{6}$$

and, by (5), the UC of $\pi_j(n)$ (the vector $\mathbf{u}_j$) are given by the element wise product of the vector $\mathbf{a}$ that solves the system (6), and the j column of $\mathbf{L}$.

If we want to use right eigenvectors, perhaps because our numerical tool only computes them, then we can proceed as follows. Let $\mathbf{R}$ be a matrix whose columns, $\mathbf{r}_1, \cdots \mathbf{r}_k$, are the right-hand eigenvectors of $(\mathbf{P})^{\mathrm{T}}$. Then, $(\mathbf{P})^{\mathrm{T}} = \mathbf{R}\,\boldsymbol{\Lambda}\,\mathbf{R}^{-1}$, and:

$$(\boldsymbol{\pi}(n))^{\mathrm{T}} = \sum_{i=1}^{k} b_i\,\lambda_i^n\,\mathbf{r}_i \tag{7}$$

Defining the column vector $\mathbf{b} = \begin{bmatrix}b_1 & \cdots & b_k\end{bmatrix}^{\mathrm{T}}$, the constants b_i can be obtained solving the system of equations:

$$\mathbf{R}\,\mathbf{b} = (\boldsymbol{\pi}(0))^{\mathrm{T}} \tag{8}$$

and, by (7), the vector $\boldsymbol{u}_j$ (see equation (4)) is given by the element wise product of the vector $\mathbf{b}$ that solves the system (8), and the j row of $\mathbf{R}$.

Note that $\mathbf{P}$ is diagonalizable only if the geometric multiplicity of all its eigenvalues is equal to its multiplicity (the

matrix is said to be *non-defective*). This is an important restriction of this method.

B. Vandermonde Method

The UC of $\pi_j(n)$, $1 \leq j \leq k$ can be obtained solving the system of equations that results from imposing the boundary conditions to equation (2):

$$\pi_j(n) = (\boldsymbol{\pi}(0)\,\mathbf{P}^n)_j \tag{9}$$

where the notation $(\boldsymbol{x})_j$ refers to the j component of the vector $\boldsymbol{x}$. Note that we need substituting in (9) for $n = 0,\ 1, \cdots$, up to the number of UC to be determined minus 1. If we do not use the fact that the geometric multiplicity of some eigenvalues may be larger that 1, then there will be k UC, where k is the number of states of the Markov chain, i.e. the size of the square matrix $\mathbf{P}$ (see remark 1). For the sake of simplicity, in the rest of the paper, if not otherwise stated, we shall obviate this fact and assume a Markov chain with k states, and k UC to be determined. Using (2) and substituting in (9) we have that the vector $\boldsymbol{u}_j$ with the UC of $\pi_j(n)$, defined by (4), can be obtained solving the system of equations:

$$\begin{bmatrix}\mathbf{A}_1 & \cdots & \mathbf{A}_L\end{bmatrix}\,\boldsymbol{u}_j = \mathbf{B} \tag{10}$$

where the sub-matrices $\mathbf{A}_l^{k\times k_l}$, $l = 1, \cdots L$, ($\lambda_1 = 0,\ \lambda_l \neq 0$) are given by:

$$\mathbf{A}_1^{k\times k_1} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \cdots & & & \end{bmatrix}$$

$$\mathbf{A}_l^{k\times k_l} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ \lambda_l & \lambda_l & \cdots & \lambda_l \\ \lambda_l^2 & 2\,\lambda_l^2 & \cdots & 2^{k_l-1}\,\lambda_l^2 \\ \cdots & & & \\ \lambda_l^n & n\,\lambda_l^n & \cdots & n^{k_l-1}\,\lambda_l^n \\ \cdots & & & \end{bmatrix} \tag{11}$$

Notice that the elements of (11) are given by: $(\mathbf{A}_l^{k\times k_l})_{ij} = (i-1)^{j-1}\,\lambda_l^{i-1}$, $1 \leq i \leq k$, $1 \leq j \leq k_l$. $\mathbf{B}$ is the column vector:

$$\mathbf{B} = \begin{bmatrix}\pi_j(0) & (\boldsymbol{\pi}(0)\,\mathbf{P})_j & (\boldsymbol{\pi}(0)\,\mathbf{P}^2)_j \cdots & (\boldsymbol{\pi}(0)\,\mathbf{P}^{k-1})_j\end{bmatrix}^{\mathrm{T}} \tag{12}$$

Note that if the chain starts in state i with probability 1 (i.e., $\boldsymbol{\pi}(0)$ is a probability vector with the probability 1 in the component i), then $\mathbf{B} = [\delta_{ij}\ \ (\mathbf{P})_{ij}\ \ (\mathbf{P}^2)_{ij} \cdots\ \ (\mathbf{P}^{k-1})_{ij}]^{\mathrm{T}}$, where δ_{ij} is the Kronecker's delta ($\delta_{ij} = 1$ for $i = j$, and 0 otherwise).

Using this approach, $\boldsymbol{u}_j$ is obtained solving a confluent Vandermonde system of equations, for which there exist abundant literature and fast numerical methods. One example is the method of Björ and Pereyra [16]. However, for large matrices this method fails to give accurate results, due to rounding errors. Therefore, in the numerical experiments given in section VII it was found more convenient building the Vandermonde matrix and solving the system using a QR decomposition [17].

This approach has the advantage over the Eigenvectors method that the matrix $\mathbf{P}$ can be defective. Therefore, this approach gives a more general solution.

Remark 2 *In order to compute the vector* $\mathbf{B}$ *it is not necessary the computation of the powers of the matrix* $\mathbf{P}$*, as it may seem from equation* (12)*. This would be costly for large matrices. Note that equation* (12) *can be implemented as the product of a row vector,* $\mathbf{r}$*,* $k-1$ *times the matrix* $\mathbf{P}$*, as shown in the algorithm III.1. Note also that if we are interested in* $\pi_j(n)$ *for* $j = \{j_1, \cdots j_n\}$*, then it is convenient to solve the system* (10) *for the matrix* $\mathbf{B}^{k\times|j|}$ *that would be computed using algorithm III.1 with the indexes* $j = \{j_1, \cdots j_n\}$*. We use the notation* $|j|$ *for the cardinality of the set* $j = \{j_1, \cdots j_n\}$*.*

Algorithm III.1 Computation of matrix $\mathbf{B}^{k\times|j|}$. We use the notation $\mathbf{v}[j]$ for a subvector of $|j|$ elements of $\mathbf{v}$ indexed by the set $j = \{j_1, \cdots j_n\}$. $\mathbf{B}[i,]$ denotes row i of matrix $\mathbf{B}$.

1: $\mathbf{B}[1,] \leftarrow \boldsymbol{\pi}_0[j]$
2: $\mathbf{r} \leftarrow \boldsymbol{\pi}_0$
3: **for all** i in $2 : k$ **do**
4: $\quad \mathbf{r} \leftarrow \mathbf{r}\,\mathbf{P}$
5: $\quad \mathbf{B}[i,] \leftarrow \mathbf{r}[j]$
6: **end for**

IV. Continuous Time Markov Chains

Let $X(t)$ be an homogeneous finite-state continuous-time Markov Chain with k states, and infinitesimal generator (rate matrix) of size k, $\mathbf{Q}^{k\times k}$. The transient solution is given by the exponential matrix $\mathrm{e}^{\mathbf{Q}\,t}$ [12]:

$$\boldsymbol{\pi}(t) = \boldsymbol{\pi}(0)\,\mathrm{e}^{\mathbf{Q}\,t},\ t \geq 0 \tag{13}$$

where $\boldsymbol{\pi}(0)$ is the initial distribution and the components of the row vector $\boldsymbol{\pi}(t)$ are the probabilities

$$\pi_j(t) = Prob\left\{X(t) = j \ \middle|\ \boldsymbol{\pi}(0)\right\},\ 1 \leq j \leq k,\ t \geq 0$$

An expression for $\pi_j(t)$ can be obtained analogously to the discrete time case.

Theorem 2 *Let* λ_l*,* $l = 1, \cdots L$ *be the eigenvalues of* $\mathbf{Q}$*, each with multiplicity* k_l *(*$k_l \geq 1$*,* $\sum_l k_l = k$*), then:*

$$\pi_j(t) = \sum_{l=1}^{L} e^{\lambda_l\,t} \sum_{m=0}^{k_l-1} a_j^{(l,m)}\,t^m,\ 1 \leq j \leq k,\ t \geq 0 \tag{14}$$

Equation (14) has been previously used in the literature [13], [14], [15], [7], and it will not be proved here.

Remark 3 *Note that, in contrast to the case of a DTMC (see theorem 1), now it is not necessary a special attention to the eigenvalue* $\lambda_l = 0$*. In fact, since* $e^{\lambda_l\,t} = 1$ *for* $\lambda_l = 0$*, the UC associated with* $\lambda_l = 0$ *in a CTMC is the only one that do not vanish in* $\pi_j(t)$ *when* $t \to \infty$*. Thus,* $\lambda_l = 0$ *is for a CTMC equivalent to what the eigenvalue* $\lambda_l = 1$ *is for a DTMC. In the rest of the paper we shall refer to the eigenvalue equal to* 1 *of a stochastic matrix* $\mathbf{P}$*, and* 0 *of a rate matrix* $\mathbf{Q}$*, as the dominant eigenvalue. Additionally, the remark 1, regarding*

to the eigenvalue $\lambda_1 = 1$ in a DTMC, has its counterpart for the eigenvalue $\lambda_1 = 0$ in a CTMC: assume that the Markov chain has the eigenvalue $\lambda_1 = 0$ with multiplicity $k_1 > 1$. Clearly, the polynomial corresponding to this eigenvalue:

$$p_j^{(1)}(t) = \sum_{m=0}^{k_1-1} a_j^{(1,m)}\, t^m \tag{15}$$

must have the coefficients $a_j^{(1,m)} = 0$, for $m = 1, \cdots, k_1 - 1$. Otherwise, the equation (14) will be unbounded (the polynomial (15) would diverge when $t \to \infty$). Therefore, for the dominant eigenvalue of both DTMC and CTMC there will be associated only 1 undetermined coefficient.

V. Undetermined Coefficients of a CTMC

In equation (14) there are up to k UCs $a_j^{(l,m)}$ of $\pi_j(t)$ to be determined. The two methods described in section III can be applied to obtain the UC of $\pi_j(t)$. The Eigenvectors method described in section III-A is almost identical for a CTMC, thus, it will not be repeated here. The only difference is that λ_i^n in equations (5) and (7), are now $\mathrm{e}^{\lambda_i\, t}$, and, in contrast to the DTMC, it is not necessary a special attention to the eigenvalue $\lambda_l = 0$. Note that for CTMCs the Eigenvectors method also requires the matrix matrix $\mathbf{Q}$ to be non-defective. This property only holds in some special cases, as in M/M/c queues. This fact has been exploited in [18] to obtain a close form transient solution for these class of queues.

A. Vandermonde Method

The UC of $\pi_j(t)$, $1 \le j \le k$ can be obtained solving the system of equations that results from imposing the boundary conditions to equation (14):

$$\frac{\partial^n \pi_j(0)}{\partial t^n} = (\boldsymbol{\pi}(0)\, \mathbf{Q}^n)_j,\ 0 \le n \le k-1 \tag{16}$$

As in the discrete case, the vector $\boldsymbol{u}_j$ with the UC of $\pi_j(t)$, defined by (4), can be obtained solving the system of equations: $\begin{bmatrix}\mathbf{A}_1 & \cdots & \mathbf{A}_L\end{bmatrix} \boldsymbol{u}_j = \mathbf{B}$, where the sub-matrices $\mathbf{A}_l^{k\times k_l}$, $l = 1, \cdots L$ are given by (see appendix A):

$$\mathbf{A}_l^{k\times k_l} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ \lambda_l & 1 & \cdots & 0 \\ \lambda_l^2 & 2\,\lambda_l & \cdots & 0 \\ \cdots & & & \\ \lambda_l^n & n\,\lambda_l^{n-1} & \cdots & \underline{n^{k_l-1}}\,\lambda_l^{n-(k_l-1)} \\ \cdots & & & \end{bmatrix} \tag{17}$$

where $\underline{n^m} = n\,(n-1)\cdots(n-m+1)$, $\underline{n^m} = 0$ for $n < m$ and $\underline{0^0} = 1$. Notice that the elements of (17) are given by: $(\mathbf{A}_l^{k\times k_l})_{ij} = \underline{(i-1)^{j-1}}\,\lambda_l^{i-j}$, $1 \le i \le k$, $1 \le j \le k_l$. $\mathbf{B}$ is the column vector:

$$\mathbf{B} = \begin{bmatrix}\pi_j(0) & (\boldsymbol{\pi}(0)\,\mathbf{Q})_j & (\boldsymbol{\pi}(0)\,\mathbf{Q}^2)_j \cdots\end{bmatrix}^{\mathrm{T}} \tag{18}$$

Note that if the chain starts in state i with probability 1 (i.e., $\boldsymbol{\pi}(0)$ is a probability vector with the probability 1 in the component i), then $\mathbf{B} = [\delta_{ij}\ (\mathbf{Q})_{ij}\ (\mathbf{Q}^2)_{ij} \cdots]^{\mathrm{T}}$. Note also that the vector $\mathbf{B}$ given by (18) has exactly the same form than in the DTMC (equation (12)). Thus, the remark 2 regarding the computation of the vector $\mathbf{B}$ for the DTMC, is applicable now changing the matrix $\mathbf{P}$ by $\mathbf{Q}$.

The method described above was proposed in [11] for a CTMC having non confluent eigenvalues. In this case, the UC are obtained solving a pure Vandermonde system of equations. Using the approach described above we do not have this limitation, thus, the method can be used to compute the transient solution of any CTMC. The only limitation is the numerical difficulties that appear in the solution of large confluent Vandermonde system of equations. This will be investigated in section VII.

VI. Uniformized Vandermonde Method

Computational round off errors may lead the CTMC solution methods described in section V to give inaccurate results, especially for chains with a large number of states. The Eigenvectors method fails, particularly for asymmetric matrices, because the numerical tool may not be able to obtain linearly independent eigenvectors (see [5]). In case of the Vandermonde method, because the eigenvalues of the matrix $\mathbf{Q}$ may be out of the unit circle, and thus, the matrix (17) used to compute the undetermined coefficients may have a large norm and be ill-conditioned. This problem can be alleviated using the well known uniformization method (see e.g. [7], [9]). This method consists of considering the *uniformized matrix*:

$$\mathbf{P} = \mathbf{I} + \frac{1}{q}\mathbf{Q} \tag{19}$$

where q is a constant such that $\max_i |(\mathbf{Q})_{ii}| \le q < \infty$, and using the equation (conveniently truncated):

$$\boldsymbol{\pi}(t) = \boldsymbol{\pi}(0)\, \mathrm{e}^{\mathbf{Q}\,t} = \sum_{n=0}^{\infty} \mathrm{e}^{-q\,t}\, \frac{(q\,t)^n}{n!}\, \boldsymbol{\pi}(0)\, \mathbf{P}^n \tag{20}$$

Note that the matrix $\mathbf{P}$ defined by (19) is stochastic, and the equation (20) can be easily proved by direct substitution of the definition of the matrix $\mathbf{P}$ (19). The problem of directly applying the uniformization formula (20) is that computing the vectors $\boldsymbol{\pi}(0)\,\mathbf{P}^n$ is costly for large number of states. Since storing the these vectors will be infeasible for large matrices, the whole series would need to be computed each time the probability for a new value of t is desired.

To cope with the above mentioned problems we can use the DTMC defined by $\mathbf{P}$ to compute the *uniformized chain probabilities* $\pi_j^{(P)}(n)$ for each of the states j we are interested, as described in section II. Thorough this section the indexes $^{(P)}$ and $^{(Q)}$ will be used to distinguish the uniformized chain and the original CTMC chain with rate matrix $\mathbf{Q}$, respectively. Note that the initial conditions, $\boldsymbol{\pi}(0)$, are the same for both chains. So, $^{(P)}$ $^{(Q)}$ will not be used with $\boldsymbol{\pi}(0)$. From equation (20) we have:

$$\pi_j^{(Q)}(t) = \sum_{n=0}^{\infty} \mathrm{e}^{-q\,t}\, \frac{(q\,t)^n}{n!}\, \pi_j^{(P)}(n) \tag{21}$$

Let $\lambda_l^{(P)}$, $l = 1, \cdots L$ be the eigenvalues of the uniformized matrix $\mathbf{P}$ given by equation (19), each with multiplicity k_l. Without loss of generality, let $\lambda_1^{(P)}$ a possible eigenvalue equal to 0, and $\lambda_2^{(P)}$ be the eigenvalue equal to 1. Let $b_j^{(l,m)}$ be the UC of $\pi_j^{(P)}(n)$. Using equation (2) and remark 1, and substituting into (21) we have:

$$\pi_j^{(Q)}(t) = \sum_{n=0}^{\infty} \mathrm{e}^{-q\,t}\,\frac{(q\,t)^n}{n!}\,\left[\sum_{m=0}^{k_1-1} b_j^{(1,m)}\delta_{n-m} + b_j^{(2,0)} + \sum_{l=3}^{L}(\lambda_l^{(P)})^n \sum_{m=0}^{k_l-1} b_j^{(l,m)}\,n^m\right] =$$
$$\mathrm{e}^{-q\,t}\left[b_j^{(1,0)} + \sum_{m=1}^{k_1-1}\frac{(q\,t)^m}{m!}\,b_j^{(1,m)}\right] + b_j^{(2,0)} + \sum_{l=3}^{L}\mathrm{e}^{-q\,t\,(1-\lambda_l^{(P)})}\left[b_j^{(l,0)} + \sum_{m=1}^{k_l-1} b_j^{(l,m)}\,q^{(l,m)}(t)\right] \quad (22)$$

where $q^{(l,m)}(t)$ are polynomials in t of degree m (see appendix A):

$$q^{(l,m)}(t) = \sum_{i=1}^{m} q_i^{(m)}\,(q\,\lambda_l^{(P)})^i\,t^i,\ l = 3, \cdots L \quad (23)$$

with the coefficients $q_i^{(m)}$ given by the recurrence relation:

$$q_i^{(m)} = \begin{cases} 1, & i = 1 \\ \sum_{k=i-1}^{m-1}\binom{m-1}{k}\,q_{i-1}^{(k)}, & i = 2, \cdots m \end{cases} \quad (24)$$

Let $\lambda_l^{(Q)}$, $l = 1, \cdots L$ be the eigenvalues of the rate matrix $\mathbf{Q}$, and $a_j^{(l,m)}$ the UC of $\pi_j^{(Q)}(t)$. Comparing equation (14) and (22), it turns out that it must be:

$$\lambda_l^{(Q)} = -q\,(1 - \lambda_l^{(P)}),\ l = 1, \cdots L \quad (25)$$

$$\lambda_l^{(P)} = 1 + \frac{\lambda_l^{(Q)}}{q},\ l = 1, \cdots L \quad (26)$$

$$a_j^{(l,0)} = b_j^{(l,0)},\ l = 1, \cdots L. \quad (27)$$

Equations (25) and (26) can be also derived directly from (19), since the eigenvalues follow the same linear transformation. From (27) we have that if there are not confluent eigenvalues, the UC of $\pi_j^{(P)}(n)$ and $\pi_j^{(Q)}(t)$ are the same. In the confluent case, and comparing again equations (14) and (22), and using (23), we have:

$$a_j^{(1,m)} = \frac{q^m}{m!}\,b_j^{(1,m)},\ m = 1, \cdots k_1 - 1 \quad (28)$$

$$a_j^{(2,0)} = b_j^{(2,0)} \quad (29)$$

$$a_j^{(l,m)} = (q\,\lambda_l^{(P)})^m \sum_{k=m}^{k_l-1} q_m^{(k)}\,b_j^{(l,k)},\quad \begin{matrix} m = 1, \cdots k_l - 1, \\ l = 3, \cdots L \end{matrix} \quad (30)$$

Note that (29) is the UC associated with the dominant eigenvalue (being $\lambda_2^{(P)} = 1$, using (25) it is $\lambda_2^{(Q)} = 0$). Thus, even if the dominant eigenvalue is not single, there is only one UC associated with it (see remark 3).

The mapping established by equations (28), (29) and (30) are of interest from a theoretical point of view. Note that these equations establish an equality between the UC of a CTMC, and the UC of one of its uniformized chains (which is a DTMC). In another words, these equations establish a mapping between a close form solution of a CTMC and a close form solution of one of its uniformized DTMCs. To the best of our knowledge, this is the first time that such mapping is established. This mapping can be used to compute the transient solution of a CTMC in terms of the solution of one of its uniformized chains, which can be convenient from a numerical point view, as explained in the following.

One could first think of using the eigenvectors method to compute the transient solution of the uniformized DTMC. However, using the definition of eigenvector it is easy to prove that equations (25) and (26) imply that the eigenvectors of $\mathbf{Q}$ and its uniformized matrix $\mathbf{P}$ given by (19), must be the same. Therefore, if the system of equations (8) is ill-conditioned for the eigenvectors of $\mathbf{Q}$, so it will be for the eigenvectors of $\mathbf{P}$. Thus, there is no much advantage in using the Eigenvectors method with the uniformized matrix $\mathbf{P}$.

On the other hand, using the Vandermonde method with the uniformized matrix $\mathbf{P}$ implies building the Vandermonde matrix using $\lambda_l^{(P)}$. This makes a big difference with respect to the Vandermonde matrix that would be obtained using $\lambda_l^{(Q)}$. In the following the Vandermonde matrices obtained using $\lambda_l^{(P)}$ and $\lambda_l^{(Q)}$ will be denoted as $\mathbf{V}^{(P)}$ and $\mathbf{V}^{(Q)}$, respectively. Notice that $\lambda_l^{(P)}$ must be inside the unit disk ($|\lambda_l^{(P)}| \leq 1$), thus, $\mathbf{V}^{(P)}$ will be possibly much better conditioned than $\mathbf{V}^{(Q)}$. Additionally, by choosing an appropriate value for q, we can tune the condition number of $\mathbf{V}^{(P)}$. So, the idea of the method is solving a CTMC by using the Vandermonde method with one of its uniformized DTMCs: one having a well conditioned matrix $\mathbf{V}^{(P)}$. Then, using the mapping given by equations (28), (29) and (30) to derive a close form solution for the original CTMC. Summing up, the *Uniformized Vandermonde Method* proposed in this paper consists of the following steps:

1) Compute the eigenvalues of $\mathbf{Q}$, $\lambda_l^{(Q)}$.
2) Process the eigenvalues $\lambda_l^{(Q)} < 0$, merging those that are *near confluent*, if any, and limiting the maximum multiplicity (see remark 5).
3) Using $\mathbf{Q}$ and its eigenvalues, choose an appropriate value for the uniformization parameter q (this point will be discussed in section VI-A).
4) Compute the uniformized matrix $\mathbf{P} = \mathbf{I} + \frac{1}{q}\mathbf{Q}$. To save memory space, the matrix $\mathbf{P}$ can overwrite matrix $\mathbf{Q}$ in this step, since $\mathbf{Q}$ is not needed anymore.
5) Using $\mathbf{P}$ and $\boldsymbol{\pi}(0)$, compute the matrix $\mathbf{B}$ using the algorithm III.1. Then, the matrix $\mathbf{P}$ can be removed, since it is not needed anymore.
6) Compute the eigenvalues of $\mathbf{P}$, $\lambda_l^{(P)}$, using equation (26).
7) Using $\lambda_l^{(P)}$, construct the Vandermonde matrix, as ex-

plained in section III-B.

8) Solve the resulting Vandermonde system to obtain the UC of $\pi_j^{(P)}(n)$. If there aren't confluent eigenvalues, these are the UC of $\pi_j^{(Q)}(t)$, and we are done. In case of confluent eigenvalues, use equations (28) and (30) to compute the UC of the corresponding confluent eigenvalues of $\pi_j^{(Q)}(t)$.
9) Use UC of $\pi_j^{(Q)}(t)$ and the eigenvalues $\lambda_l^{(Q)}$ in equation (14) to evaluate the desired transient solution.

Remark 4 *It is interesting to note that if the uniformized chain probabilities $\pi_j^{(P)}(n)$ given by the matrix $\mathbf{P}$ (19) have a limiting distribution, they converge to $\pi_j^{(Q)}(t)$ at the points $n = [q\,t]$, where $[x]$ stands for the integral value of x. This comes from the fact:*

$$\lim_{n\to\infty} \mathbf{P}^n = \lim_{n\to\infty} \left(\mathbf{I} + \frac{1}{q}\mathbf{Q}\right)^n = \lim_{n\to\infty} \left(\mathbf{I} + \frac{t\,\mathbf{Q}}{n}\right)^n = e^{\mathbf{Q}\,t} \tag{31}$$

Remark 5 *Having eigenvalues of $\mathbf{P}$ near confluent will make the matrix $\mathbf{V}^{(P)}$ to have repeated rows (due to rounding errors), and thus, be ill-conditioned. By near confluent we mean eigenvalues $\lambda_a^{(P)}$ and $\lambda_b^{(P)}$ such that $|\Re\lambda_a^{(P)} - \Re\lambda_b^{(P)}| < \epsilon$ and $|\Im\lambda_a^{(P)} - \Im\lambda_b^{(P)}| < \epsilon$, where $\Re$ and $\Im$ are the real and imaginary part respectively, and ϵ is a small positive number. To solve this problem it has been proceed as follows: first, all eigenvalues satisfying the above inequality, have been merged (i.e. considered as the same eigenvalue, increasing its multiplicity). For the same reason, all complex conjugate eigenvalues pairs having $|\Im\lambda_b^{(P)}| < \epsilon$ have also been merged, by removing their imaginary part. In the numerical experiments of section VII it was found convenient to use $\epsilon = ((1+\epsilon_m)^k - 1)^{0.5}$, where ϵ_m is the smallest positive floating-point number such that $1 + \epsilon_m > 1$ and k is the number of rows of $\mathbf{V}^{(P)}$.*

Another problem may arise in case of having eigenvalues with large multiplicity. Due to rounding errors, this may happen even if the exact eigenvalues of the matrix are all single. This could produce huge elements in $\mathbf{V}^{(P)}$ (see equation (11)), and the system would become ill-conditioned. To cope with this problem, the multiplicity of the eigenvalues has been limited to a maximum value M. In other words, if the multiplicity k_l of an eigenvalue $\lambda_l^{(P)}$ is $k_l > M$, then it is assumed that the UC $a_j^{(l,m)} = 0$ for $m = M, \cdots k_l - 1$. Note that this is equivalent to assume that the geometric multiplicity of the eigenvalue is $> k_l - M$ (see remark 1), which in many cases may be true. An additional advantage of doing so, is that the size of the Vandermonde system to solve is reduced. In the numerical experiments of section VII it has been set $M = 5$.

A. Choosing the uniformization parameter

A critical point in the UV method is the selection of the uniformization parameter q. When the uniformization equation (20) is used, it is common to take the uniformization parameter $q = \max_i |(\mathbf{Q})_{ii}|$ [9]. This is the minimum value that q can have for the matrix $\mathbf{P}$ to be stochastic. However, in order improve the condition number of the Uniformized Vandermonde matrix, $\mathbf{V}^{(P)}$, it may be better choosing a higher value for q (note from (26) that taking $q \to \infty$, all eigenvalues $\lambda_l^{(P)} \to 1$). First, it has been observed in the numerical experiments that $\mathbf{V}^{(P)}$ is better conditioned having all the elements in the non negative x-plane. This can be explained by the fact that the inaccuracies due rounding errors are lower if the real part of all the elements of the matrix have the same sign. From equation (26) we have that this goal can be achieved if $q \geq |\min_l \Re\lambda_l^{(Q)}|$, where $\min_l \Re\lambda_l^{(Q)}$ is the most negative real part of the eigenvalues of $\mathbf{Q}$. Additionally, if all the eigenvalues $|\lambda_l^{(P)}| < 1$ are close to the origin, then the terms $(\lambda_l^{(P)})^n$ of $\mathbf{V}^{(P)}$ will vanish when increasing n, and $\mathbf{V}^{(P)}$ will be ill-conditioned. To solve this problem we can proceed as follows. Let k be the number of rows of $\mathbf{V}^{(P)}$, and $\lambda_m^{(P)}$ the second largest eigenvalue, in modulus (recall that the largest is $\lambda_1^{(P)} = 1$). A rule of thumb is choosing q such that $|\lambda_m^{(P)}|^k > \epsilon$, for some small positive ϵ. In the numerical experiments of section VII it was used $\epsilon = \epsilon_m$, where ϵ_m is the smallest positive floating-point number such that $1 + \epsilon_m > 1$. From (26) we have that $\lambda_m^{(P)} = 1 + \frac{\lambda_m^{(Q)}}{q}$, where $\lambda_m^{(Q)}$ is the second smallest eigenvalue of $\mathbf{Q}$, in modulus (the smallest is $\lambda^{(Q)} = 0$). Thus, we have that it must be: $q > |\lambda_m^{(Q)}|/(1-\epsilon^{1/k})$. Putting all together, the uniformization parameter used in the numerical experiments of section VII has been chosen as:

$$q = \max\left\{\max_i |(\mathbf{Q})_{ii}|,\ |\min_l \Re\lambda_l^{(Q)}|,\ \frac{|\lambda_m^{(Q)}|}{1-\epsilon^{1/k}}\right\} \tag{32}$$

VII. Numerical Experiments

In this section three different methods to compute the transient solution of a CTMC with rate matrix $\mathbf{Q}$ are compared. The following notation is used:

- *qevec*: Eigenvectors method (see section V).
- *qvand*: Vandermonde method, (see section V-A).
- *uvand*: Uniformized Vandermonde method, (see section VI).

The results have been computed using the R numerical tool [19] version 2.11.1, with its internal lapack and blas libraries. The experiments were done in a PC with a 64 bits Intel Xeon Dual-Core 2.3 GHz, and 12 GB of RAM.

The experiments have been done using an M/M/1/N queue, where $N \geq 1$ is the system capacity. This queue has also been used by other authors in similar numerical experiments [20], [7]. Several reasons justify this choice. First, because it is one of the few queues for which there exist close formulas for the transient solution. Therefore, we can evaluate the accuracy of the methods. Additionally, varying the system parameters we can easily change the structure of the Q matrix, and thus, analyze the generally of the method.

The state of the Markov chain is the number in the system (so, thorough this section $\pi_j(t)$, $0 \leq j \leq N$ is the probability of the queue having j customers in the system at time t),

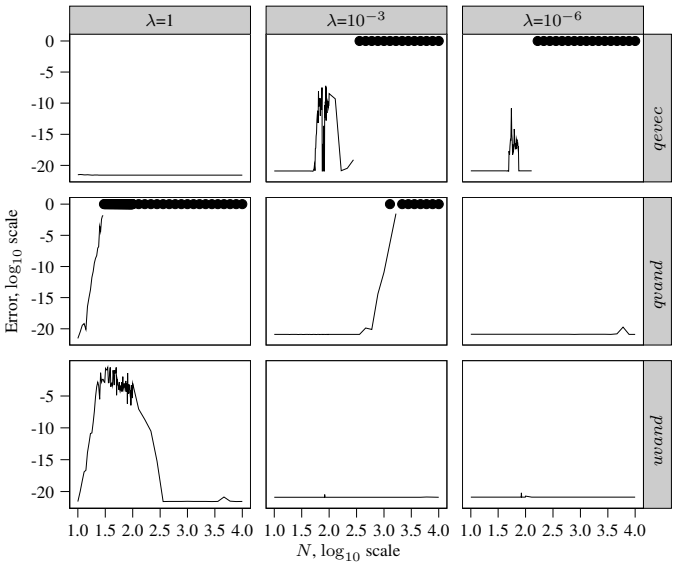

Figure 1. Error.

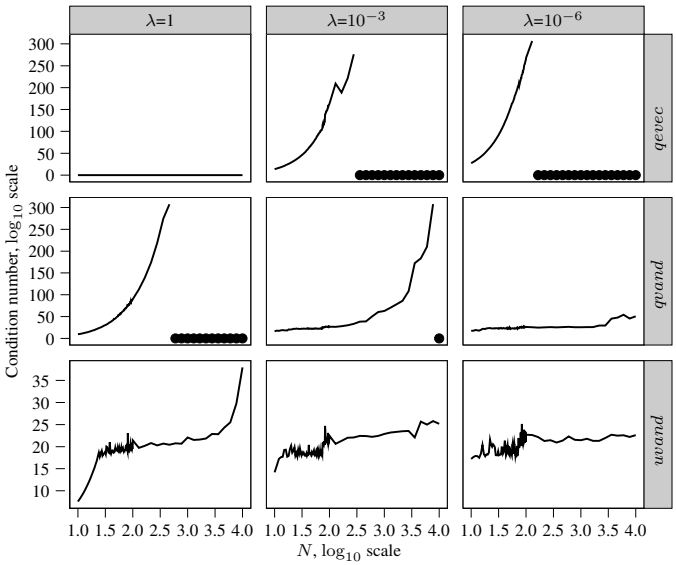

Figure 2. Condition number.

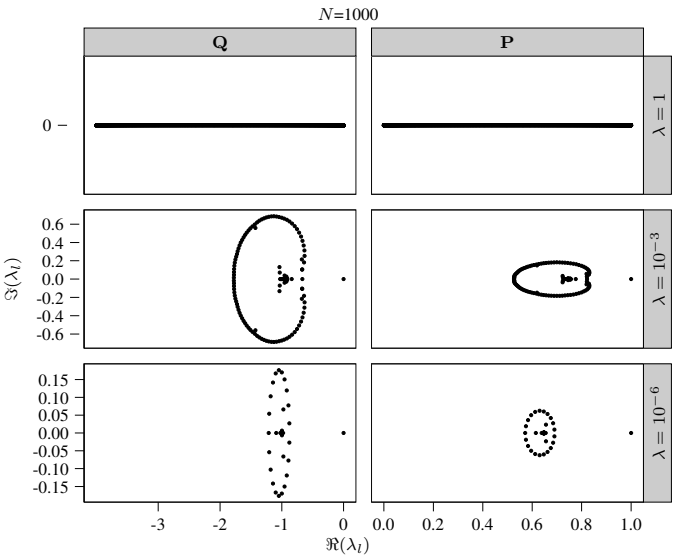

Figure 3. Complex-plane plot of the eigenvalues of the matrices $\mathbf{Q}$ and $\mathbf{P}$ for $N = 1000$.

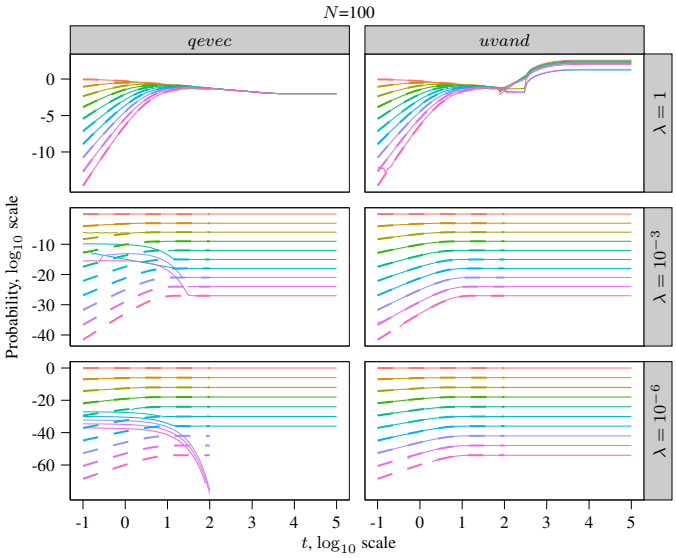

Figure 4. Probabilities $\pi_j(t)$, $j = \{0, \cdots 9\}$ for the *qevec* and *uvand* methods for $N = 100$. Values obtained with Sharma's formula are plotted with dashed lines.

and we assume that the initial state is $j = 0$. The system capacity (N), has been varied between 10 and 10^4 (note that the number of states of the Markov chain is $N + 1$). In the interval $[10, 100]$ the transient solution was computed for all values of N, since each computation took few seconds. In the interval $(100, 10^4]$ there were taken 20 points evenly spaced in log scale (10 per decade). The service rate has been set to $\mu = 1$ and the arrival rate have been set to three different values: $\lambda = 1$, 10^{-3} and 10^{-6}.

The methods are compared against the formula proposed by Sharma and Tarabia in [21]. Due to the combinatorial coefficients of Sharma's formula, its numerical evaluation gives very accurate results for values of t up to 100 s approximately.

In order to estimate the error for each of the methods $m = \{$*qevec, qvand, uvand*$\}$ it has been proceed as follows. $\pi_0^{(m)}(t)$ have been evaluated at 60 values of t evenly spaced in log scale in the range $(10^{-1}, 10^2]$ (20 points per decade). All values $\pi_0^{(m)}(t) < 0$ and $\pi_0^{(m)}(t) > 1$ where considered as failed. If the number of failed points was larger than 12 (more than 20% of failed points), it was considered that the method failed. For the non-failed points, n, it was computed:

$$\text{error}^{(m)} = \frac{1}{n} \sum_{i=1}^{n} \frac{(\pi_0^{(m)}(t_i) - \pi_0^{(sh)}(t_i))^2}{\pi_0^{(sh)}(t_i)} \qquad (33)$$

where $\pi_0^{(sh)}$ is the probability obtained with Sharma's formula.

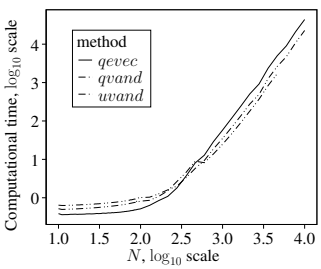

Figure 5. Computational time.

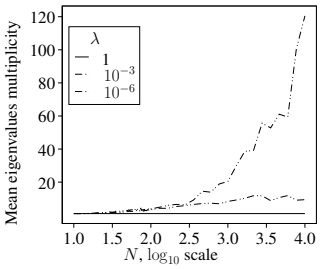

Figure 6. Mean eigenvalues multiplicity.

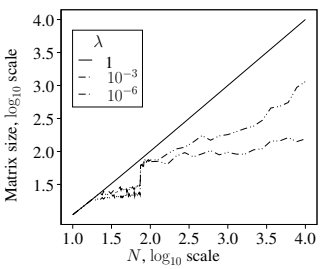

Figure 7. $\mathbf{V}^{(P)}$ matrix size in the $uvand$ method.

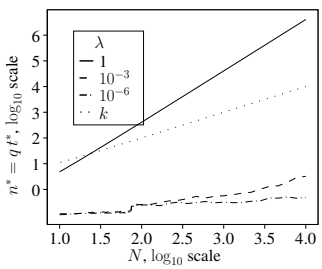

Figure 8. Values $n^* = q\,t^*$, and $k = N + 1$.

Figure 1 shows the error obtained in each scenario. The failed points, or those points where the solver failed to compute the UC (because rounding errors made the numerical tool finding the matrix singular), are marked with a dot at error=1. Figure 2 shows the condition number of the matrix used to computed the UC: the Eigenvectors matrix for $qevec$, and the Vandermonde matrices $\mathbf{V}^{(Q)}$ and $\mathbf{V}^{(P)}$ for $qvand$ and $uvand$ respectively (see section VI). In figure 2 the points located at $y = 1$ mark the scenarios where the solver failed to compute the UC. Both figures are in log-log scale.

Figure 1 shows that $qevec$ is very accurate for $\lambda = 1$. This is because $\mathbf{Q}$ is symmetric for $\lambda = 1$, and it is known that Eigenvectors method works very well in this case (see e.g. [5]). In fact figure 2 shows that the condition number for $\lambda = 1$ in $qevec$ is constant and equal to 1. For $\lambda \neq 1$ figure 2 shows that the condition number for $qevec$ increases rapidly with increasing N. In fact, the lower is λ, the less symmetric is $\mathbf{Q}$, and the smaller are the values of N for which $qevec$ is able to computed the UC. Regarding the method $qvand$, figure 2 shows that the condition number increases rapidly with N for $\lambda = 1$ and 10^{-3}, explaining the bad results observed for this method in figure 1 for these values of λ. This is because the eigenvalues of $\mathbf{Q}$ are out of the unit circle, and thus, the norm of $\mathbf{V}^{(Q)}$ increases rapidly with increasing N. This can be observed in figure 3. This figure shows a complex-plane plot of the eigenvalues obtained with the numerical tool for the matrix $\mathbf{Q}$, and the uniformized matrix $\mathbf{P}$, for $N = 1000$.

Finally, figure 1 shows that $uvand$ is the only method that succeeds to compute the transient solution for all values of λ and N. Figure 2 shows that the condition number of $\mathbf{V}^{(P)}$ is between 10^{20} and 10^{25} for most of the values of N. This is a large condition number, however, figure 1 shows that the error for $uvand$ is very low when $\lambda \neq 1$.

To explain what happens with $uvand$ when $\lambda = 1$, figure 4 depicts the probabilities $\pi_j^{(m)}(t)$, $j = 0, \cdots 9$ for the scenarios $m = \{$*qevec, uvand*$\}$ and $\lambda = \{1,\ 10^{-3},\ 10^{-6}\}$ when $N = 100$. In the interval $t \in [10^{-1},\ 10^{2}]$ the values obtained with Sharma's formula are plotted with dashed lines (note that the error depicted in figure 1 was computed in this interval). Figure 4 shows that $\pi_j^{(uvand)}(t)$ starts diverging when t is approximately 100 s. In fact, the error observed in figure 1 in $uvand$ method for N between 10 and $10^{2.5}$, is because there were taken some samples of $\pi_j^{(uvand)}(t)$ in the region where it diverges. For $N > 10^{2.5}$ the error is negligible because the divergence of $\pi_j^{(uvand)}(t)$ starts after the interval considered in the error estimation. This instability is discussed in section VIII.

Figure 5 shows the computation time in seconds for each method in the scenario with $\lambda = 1$, for the points where the numerical tool was able to compute the UC. It can be seen that for values of N up to $10^{2.5}$, approximately, $qevec$ is the fastest method. However, for larger values of N the methods $qvand$ and $uvand$ are faster. This is motivated by the cost of computing the eigenvectors in the $qevec$ method.

An additional advantage of the $uvand$ method is the reduction of the required UC in some scenarios. This is shown in figures 6 and 7, which respectively plot the mean multiplicity of the eigenvalues computed by the numerical tool, and the $\mathbf{V}^{(P)}$ matrix size (number of required UC, k, in the $uvand$ method). In the extreme case ($\lambda = 10^{-6}$ and $N = 10^4$) figure 6 shows that the numerical tool yielded a mean multiplicity of the eigenvalues equal to 120.5. Figure 7 shows that only 155 different UC were needed to compute the transient solution for this scenario. This result comes from the fact that the maximum multiplicity was limited to $M = 5$ (see remark 5),

VIII. Discussion

The intuitive explanation of the divergence of $\pi_j^{(uvand)}(t)$ when $\lambda = 1$ is the following. The Uniformized Vandermonde method can be interpreted as a fitting of the points of the uniformized probabilities, $\pi_j^{(P)}(n)$, to the samples $\mathbf{P}^n$, for $n = 0, 1, \cdots k-1$. In the interval where these samples are taken, the corresponding probabilities obtained for $\pi_j^{(Q)}(t)$ are very accurate (recall from 4 that $\pi_j^{(P)}(n)$ converges to $\pi_j^{(Q)}(t)$ at the points $n = q\,t$). If $\pi_j^{(P)}(n)$ reaches the stationary distribution within the *sampling interval* $n = 0, 1, \cdots k-1$, then $uvand$ is very accurate for all values of t. This comes from the fact that the UC would have been computed taking into account the evolution of the chain up to the stationary regime. This fact holds for $\lambda = \{10^{-3},\ 10^{-6}\}$, but for $\lambda = 1$ the probabilities $\pi_j^{(Q)}(t)$ converge to the stationary regime very slowly. Note that this is an expected result, since an M/M/1 queue is null recurrent for $\rho = \lambda/\mu = 1$. Thus, for $N \to \infty$ the duration of the transient regime $\to \infty$. We conclude that the method is expected to give accurate results as far the duration

of the transient regime of the chain is not much larger than the interval used to compute the UCs.

In order to check this condition it can be proceed as follows. The duration of the transient regime of $\pi_j^{(Q)}(t)$ is related with the second smallest eigenvalue, in modulus: $\lambda_l^* = \min_{l,\lambda_l \neq 0} |\lambda_l^{(Q)}|$. If λ_l^* is close to 0, then $\mathrm{e}^{-\lambda_l^* t}$ will vanish for large t, thus, with a long *transient* regime. Define

$$n^* = q\, t^* = \frac{\alpha\, q}{\min_{l,\lambda_l \neq 0} |\lambda_l^{(Q)}|} \tag{34}$$

Note that $\mathrm{e}^{-\lambda_l^* t^*} = \mathrm{e}^{-\alpha}$. Thus, t^* is an estimation of the time where $\pi_j^{(Q)}(t)$ reaches the its stationary regime. Note that at time t^* all coefficients of $\pi_j^{(Q)}(t)$, but the stationary one, have vanished by at least $\mathrm{e}^{-\alpha}$. Figure 8 depicts the values of $n^* = q\, t^*$ given by equation (34), for $\alpha = 0.1$, and $k = N+1$. The figure confirms that $\pi_j^{(Q)}(t)$ converges very fast to its stationary regime for $\lambda = \{10^{-3},\ 10^{-6}\}$, but very slowly for $\lambda = 1$. For instance, when $N = 100$, $n^* = \{413,\ 0.26,\ 0.25\}$ for $\lambda = \{1,\ 10^{-3},\ 10^{-6}\}$, respectively. In fact, figure 8 shows that for $\lambda = 1$, the duration of the transient regime becomes larger than k for around $k = 25$.

We conclude with the following rule to estimate the stability of the method: Compute n^* given by (34). If $n^* \leq k$ (recall the k is the number of UC to be determined), then method is expected to give a very accurate close formula for $\pi_j^{(Q)}(t)$. Otherwise, $\pi_j^{(Q)}(t)$ may diverge when for values of $t \gg k/q$.

The divergence problem observed for the $uvand$ method claims for discussion. First note that, due to rounding errors, none method is completely stable [5]. The best method depends on the properties of the rate matrix. This comes clear from the numerical results, where the eigenvectors method was always stable for $\lambda = 1$, but failed rapidly otherwise. Methods based on matrix series, as the uniformization method, have shown to be very robust, and may be more general than the method proposed in this paper. Nevertheless, the close formula provided by the uvand method has important advantages. For instance, once the eigenvalues and the UCs have been computed, it can be evaluated much faster than a matrix series. Additionally, it can save memory, since at most $2\,N$ values need to be stored (the eigenvalues and the UCs), instead of the N^2 values of the rate matrix.

IX. Conclusions

In this paper are investigated the class of methods based on the *undetermined coefficients* approach to compute a close form transient solution of continuous time Markov chains (CTMCs). This approach consists on guessing the solution in terms of a set of undetermined coefficients (UC), and derive a system of equations that computes them. The well known Eigenvectors method belongs to this class, but it has the drawback that it cannot be used to solve defective matrices, which are likely to occur in practice due to rounding errors.

Another approach consists of applying the undetermined coefficients approach directly to the transient solution of a Markov chain in terms of its eigenvalues. Doing this way it is obtained a Vandermonde system of equations, and we refer to it as the *Vandermonde Method*. This method has the advantage that it can be used even for chains having a defective matrix. However, the Vandermonde system of equations obtained for CTMCs may be ill-conditioned for chains having a large number of states.

To solve this problem, a mapping is derived between a close form transient solution of a CTMC, and a close form transient solution of one of its uniformized discrete time Markov chains (DTMCs). We exploit this mapping by proposing a new approach to compute the transient solution of CTMCs by using the Vandermonde method with one of its uniformized DTMCs: one having a well conditioned matrix. We refer to this approach as the *Uniformized Vandermonde* (UV) method. The benefits of this new method are analyzed through extensive numerical results. The method is simple to implement and numerical results show that it is rather general, giving an accurate solution in most scenarios where it was tested. Furthermore, a rule of thumb is given to estimate when the method may diverge.

Appendix

A. Derivation of $\frac{\partial^n \pi_j(0)}{\partial t^n}$

In order to solve the system of equations (16) we need to compute the derivatives of functions of the type:

$$\pi_{k-1}(t) = (a_0 + a_1\, t + \cdots + a_{k-1}\, t^{k-1})\, \mathrm{e}^{\lambda t} \tag{35}$$

In this appendix a general formula is obtained. This is necessary to implement a script for solving the system of equations given by (16). Start by noting that:

$$\begin{aligned}\frac{\partial \pi_{k-1}(t)}{\partial t} = [&(\lambda a_0 + a_1) + (\lambda a_1 + 2\, a_2)\, t\ + \cdots \\ &(\lambda a_{k-2} + (k-1)\, a_{k-1})\, t^{k-2} + \lambda a_{k-1}\, t^{k-1})]\, \mathrm{e}^{\lambda t}\end{aligned} \tag{36}$$

which is again a polynomial in t of degree $(k-1)$ times an exponential. Let identify the coefficients of the polynomial associated with $\frac{\partial^n \pi_{k-1}(t)}{\partial t^n}$ by $a_m(n)$, $m = 0, \cdots (k-1)$, and define the column vector $\mathbf{a}(n) = \begin{bmatrix} a_0(n) & \cdots & a_{k-1}(n) \end{bmatrix}$. Note that $\mathbf{a}(0) = \begin{bmatrix} a_0 & \cdots & a_{k-1} \end{bmatrix}$. By induction it can be easily obtained that $\mathbf{a}(n)$ obeys the systems of linear difference equations $\mathbf{a}(n+1) = \mathbf{A}\, \mathbf{a}(n)$, where:

$$\mathbf{A}^{k\times k} = \begin{bmatrix} \lambda & 1 & 0 & 0 & \cdots \\ 0 & \lambda & 2 & 0 & \cdots \\ 0 & 0 & \lambda & 3 & \cdots \\ \cdots & & & & \end{bmatrix} \tag{37}$$

This system has the solution $\mathbf{a}(n) = \mathbf{A}^n\, \mathbf{a}(0)$. Writing $\mathbf{A} = \lambda\, \mathbf{I} + \mathbf{U}$, where $\mathbf{U}$ is the matrix:

$$\mathbf{U}^{k\times k} = \begin{bmatrix} 0 & 1 & 0 & 0 & \cdots \\ 0 & 0 & 2 & 0 & \cdots \\ 0 & 0 & 0 & 3 & \cdots \\ \cdots & & & & \end{bmatrix} \tag{38}$$

we have that:

$$\mathbf{A}^n = \sum_{m=0}^{n} \binom{n}{m} \lambda^{n-m}\, \mathbf{U}^m \tag{39}$$

where $\mathbf{U}^0 = \mathbf{I}$ and $\mathbf{U}^m$ is a matrix with the m upper diagonal equal to:

$$m!, \frac{(m+1)!}{1!}, \frac{(m+2)!}{2!}, \overset{n-k}{\cdots}, \frac{(k-1)!}{(k-m-1)!} \tag{40}$$

for $m = 1, \cdots k-1$, and $\mathbf{U}^m = \mathbf{0}$ for $m \geq k$. Note now that:

$$\frac{\partial^n \pi_{k-1}(0)}{\partial t^n} = a_0(n) \tag{41}$$

which is the first component of the vector $\mathbf{a}(n)$. Thus, using (39) and (40) we obtain:

$$\frac{\partial^n \pi_{k-1}(0)}{\partial t^n} = \sum_{m=0}^{k-1} \binom{n}{m} m!\, \lambda^{n-m}\, a_m = \sum_{m=0}^{k-1} n^{\underline{m}}\, \lambda^{n-m}\, a_m \tag{42}$$

where $n^{\underline{m}} = n\,(n-1)\cdots(n-m+1)$, i.e. m-permutations of n, with $n^{\underline{m}} = 0$ for $n < m$, and $0^{\underline{0}} = 1$.

Let $c_j^{(l,m)}(n)$ be the coefficient of the equation n, $0 \leq n \leq k-1$ of the system of equations (16) that multiplies the UC $a_j^{(l,m)}$ of $\pi_j(t)$. Using (42) we conclude that:

$$c_j^{(l,m)}(n) = n^{\underline{m}}\, \lambda_l^{n-m} \tag{43}$$

B. Derivation of $q^{(l,m)}(t)$

In equation (22) we need to compute the following summation for the eigenvalues with multiplicity higher than 1:

$$S_m = \sum_{n=0}^{\infty} \frac{(q\,t)^n}{n!}\, (\lambda_l^{(P)})^n\, n^m = \sum_{n=0}^{\infty} \frac{x^n}{n!}\, n^m,\ m \geq 1 \tag{44}$$

where $x = q\,t\,\lambda_l^{(P)}$. Clearly, $S_0 = \mathrm{e}^x$ (assuming $n^m = 1$, $\forall n$ for $m = 0$) and,

$$\begin{aligned} S_m &= \sum_{n=0}^{\infty} \frac{x^n}{n!}\, n^m = x \sum_{n=1}^{\infty} (n-1+1)^{m-1} \frac{x^{n-1}}{(n-1)!} \\ &= x \sum_{n=1}^{\infty} \sum_{i=0}^{m-1} \binom{m-1}{i} (n-1)^i \frac{x^{n-1}}{(n-1)!} = \\ &\qquad x \sum_{i=0}^{m-1} \binom{m-1}{i} S_i,\ m \geq 1 \end{aligned} \tag{45}$$

From (45) it can be easily derived that S_m are $\mathrm{e}^{q\,t\,\lambda_l^{(P)}}$ times polynomials in t of degree m: $S_m = \mathrm{e}^{q\,t\,\lambda_l^{(P)}}\, q^{(l,m)}(t)$, where:

$$q^{(l,m)}(t) = \sum_{i=1}^{m} q_i^{(m)}\, (q\,\lambda_l^{(P)})^i\, t^i \tag{46}$$

with coefficients $q_i^{(m)}$ given by the recurrence relation:

$$q_i^{(m)} = \begin{cases} 1, & i = 1 \\ \sum_{k=i-1}^{m-1} \binom{m-1}{k}\, q_{i-1}^{(k)}, & i = 2, \cdots m \end{cases} \tag{47}$$

References

[1] R. A. Marie, A. L. Reibman, K. S. Trivedi, Transient analysis of acyclic Markov chains, Performance Evaluation 7 (3) (1987) 175–194.

[2] G. Hjálmtỳsson, W. Whitt, Periodic load balancing, Queueing systems 30 (1) (1998) 203–250.

[3] P. Heegaard, K. S. Trivedi, Network survivability modeling, Computer Networks 53 (8) (2009) 1215–1234.

[4] A. Ramesh, K. S. Trivedi, Semi-numerical transient analysis of markov models, in: ACM-SE 33: Proceedings of the 33rd annual on Southeast regional conference, ACM, 1995, pp. 13–23.

[5] C. Moler, C. Van Loan, Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later, SIAM review 45 (1) (2003) 3–49.

[6] D. Dormuth, A. Alfa, Two finite-difference methods for solving MAP(t)/PH(t)/1/K queueing models, Queueing systems 27 (1) (1997) 55–78.

[7] A. L. Reibman, K. S. Trivedi, Numerical transient analysis of markov models, Comput. Oper. Res. 15 (1) (1988) 19–36.

[8] H. Nabli, Transient and asymptotic analysis of general markov fluid models, Queueing Systems 47 (3) (2004) 283–304.

[9] W. J. Stewart, Introduction to the numerical solution of Markov chains, Princeton University Press, 1994.

[10] S. Elaydi, An Introduction to Difference Equations, 3rd Edition, Springer, 2005.

[11] M. Vidyasagar, A novel method of evaluating e^{At} in closed form, Automatic Control, IEEE Transactions on 15 (5) (1970) 600 – 601. doi:10.1109/TAC.1970.1099568.

[12] S. Karlin, H. Taylor, A first course in stochastic processes, Vol. 697, Academic press New York, 1975.

[13] W. H. Jr, J. Fillmore, D. Smith, Matrix Exponentials: Another Approach, SIAM review 43 (4) (2001) 694–706.

[14] I. Leonard, The matrix exponential, SIAM Rev. 38 (3) (1996) 507–512.

[15] J. A. Sjogren, Closed-form solution of decomposable stochastic models, Computers & Mathematics with Applications 23 (12) (1992) 43–67.

[16] Å. Björck, V. Pereyra, Solution of vandermonde systems of equations, Mathematics of Computation 24 (112) (1970) pp. 893–903.

[17] G. Stewart, The Decompositional Approach to Matrix Computation, Computing in Science & Engineering 2 (1) (2002) 50–59.

[18] W. Grassmann, J. Tavakoli, Transient solutions for multi-server queues with finite buffers, Queueing Systems 62 (1) (2009) 35–49.

[19] R Development Core Team, R: A Language and Environment for Statistical Computing, R Foundation for Statistical Computing, Vienna, Austria, ISBN 3-900051-07-0 (2010).
URL http://www.R-project.org/

[20] M. Malhotra, J. K. Muppala, K. S. Trived, Stiffness-tolerant methods for transient analysis of stiff markov chains, Microelectronics and Reliability 34 (1994) 1825–1841.

[21] O. Sharma, A. Tarabia, A Simple Transient Analysis of an M/M/1/N Queue, The Indian Journal of Statistics (2000) 273–281.

[22] W. J. Stewart, MARCA Models: A Collection of Markov Chain Models. URL http://www4.ncsu.edu/~billy/MARCA_Models/MARCA_Models.html

[23] W. Feller, An Introduction to Probability Theory and Its Applications, Vol. 1, 3rd Edition, Wiley, 1968.